Available online at www.sciencedirect.com
SciVerse ScienceDirect
Procedia Engineering 38 (2012) 2103 – 2110
Procedia Engineering
www.elsevier.com/locate/procedia

International Conference On Modeling, Optimization And Computing (ICMOC2012)

Pose Invariant Hand Biometry Using Multiple Database

Neena Susan Alex[a], E.S. Karthik Kumar[b], Jose J Tharayil[a],a*

[a]M. Tech Embedded System, Hindustan Institute of Technology & Science, Chennai, India.
[b]Asst. Prof. Electronics & Communication dept., Hindustan Institute of Technology & Science, Chennai, India

Abstract

We propose the significance in use of multiple databases for the purpose of authentication using the palm-print of the hand for increased speed. The technique is used for authentication using the acquired image of the hand. The hand could be held in any pose hence providing pose independence. We use Appearance analysis for feature extraction. This technique also suggests hand independence for biometric authentication. It also provides a robust technique for detecting the extremities using a neighborhood scan method.

Keywords: Neighborhood method; palm-print; valley-peak points; Pose correction;

1. Introduction

The evolution of hand biometry can be classified into four generations. Constrained and contact based [1], [2].In the beginning when the hand geometry was introduced, the sensors were contact based; it also used pegs and other constraints to held the hand in position. Unconstrained and contact based [5], [6], [7], [9].

Here the hand geometry was developed to do away with pegs and other constraints. Unconstrained contact free and position dependent Contact free [3], [8], [10], [11] technique was developed for hand biometry, but the disadvantage for the developed technique is that it assumes that the person held his hand perfectly parallel to the camera or sensor. Using either geometric or palm print information. For identifying hand biometry either geometric or palm print information used and no techniques specify the use of both for identification. In all of the previous work either left or right hand was used, not both.

For hygiene, increased security and ease of use people generally prefer contact free technique for biometric identification/ Authentication. For hand biometry people may wish to use either right or left

doi:10.1016/j.proeng.2012.06.252

hand. By separating out the right or left hand and creating separate databases for each increases the speed of hand biometry

Recently more emphasis is given for biometric securities. One of the main challenges in using the hand biometry is the pose of the hand. The pose of the hand will be varying on each use. The solution for this is pose correction. Principle component analysis is a good technique that can be used for feature extraction. Euclidean distance and curvature properties can be used for extracting the valley and peak points of the hand efficiently.

2. Methodology

\

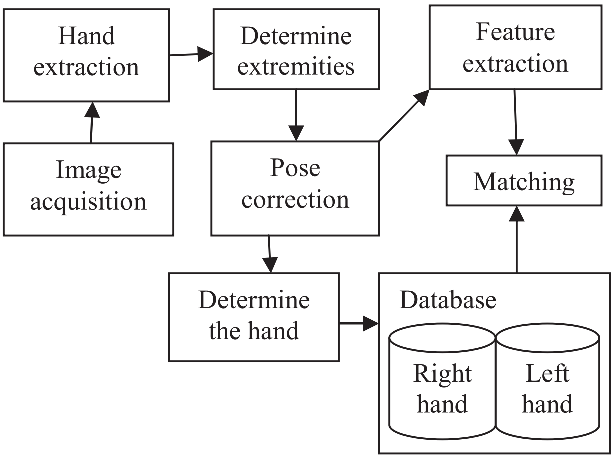

Figure i: General Block Diagram

Subtracting or absorbing certain wavelengths of color while reflecting other wavelengths back to the viewer. The offset printing process

2.1. Hand Image Acquisition

For image acquisition any digital cameras can be employed. All the images are captured with homogeneously colored background, this helps in extraction of the hand much easier. The images are taken in a well-lit environment. The fingers must be kept separately for the acquisition so that the extremities can be easily found.

2.2. Hand Extraction

The easiest approach in hand extraction is Otsu's method [4] which separates the background image from the foreground image.

2.3. Extracting valley points

The best technique for finding the extremities is by determining the curvature of the hand [12] for this

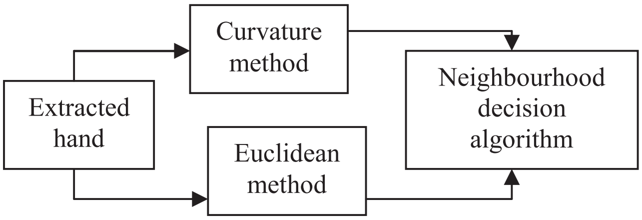

Figure ii:Block diagram of Neighbourhood Method

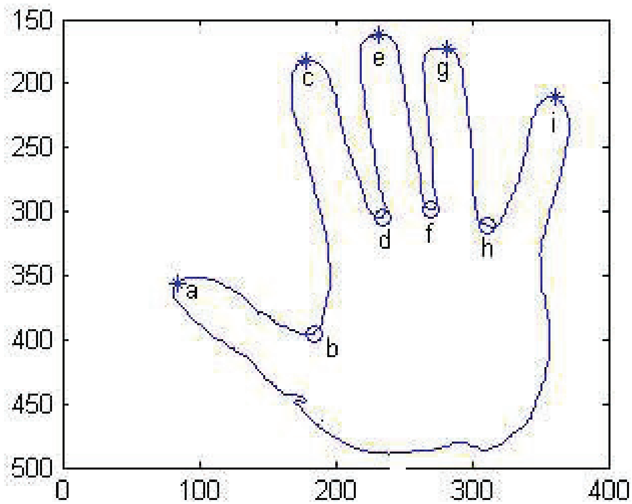

Figure iii: Figure showing various valley and peak points

1. Detect edges using the Sobel edge detector to obtain a binary edge map.
2. Extract edge contours from the edge map. When the edge reaches an end point, fill the gap and continue the extraction if the end point is nearly connected to another end point, or mark this point as a T-junction corner if the end point is nearly connected to an edge contour, but not to another end point
3. After contour extraction, compute the curvature at a fixed low scale for each contour to retain the true corners, and regard the local maxima of absolute curvature as corner candidates.
4. Compute a threshold adaptively according to the mean curvature within a region of support. Round corners are removed by comparing the curvature of corner candidates with the adaptive threshold.
5. Based on a dynamically recalculated region of support, evaluate the angles of the remaining corner candidates to eliminate any false corners.
6.

Curvature is computed using

$$K_i^j = \frac{\Delta X_i^j \Delta^2 Y_i^j - \Delta^2 X_i^j \Delta Y_i^j}{[(\Delta X_i^j)^2 + (\Delta Y_i^j)^2]} \quad \text{for } i=1,2,\ldots\ldots,N, \qquad \text{(i)}$$

Where $\Delta X_i^j = (X_{i+1}^j - X_{i-1}^j)/2$, $\Delta Y_i^j = (Y_{i+1}^j - Y_{i-1}^j)/2$

$$\Delta^2 X_i^j = (\Delta X_{i+1}^j - \Delta X_{i-1}^j)/2 \,, \Delta^2 Y_i^j = (\Delta Y_{i+1}^j - \Delta Y_{i-1}^j)/2 \qquad \text{(ii)}$$

X_i^j, Y_i^j are the coordinates of the *i*'th pixel on the *j*'th contour

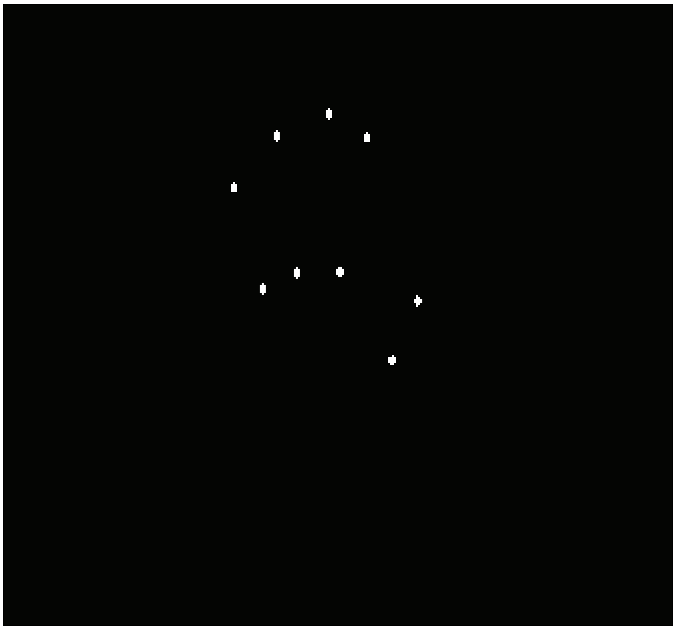

Figure iv: Extracted extremities

A more robust alternative technique was provided by Euclidean distance. Hamming distance across the rows is first calculated. The highest hamming distance is provided by a line following the centre of the hand. The image area below this line corresponding to the hamming distance is removed. Then filling is done. Filling is nothing but making all the pixels below any white pixel in the image white. This helps in finding error free Euclidean distance.

Then edge is extracted using Sobel transform. Then Euclidean distance with respect to the base of the image is calculated. The resulting sequence of radial distances yields minima and maxima corresponding to the sought extreme points. The maxima and minima are extracted from the image by dividing the graph into various segments. The resulting maxima and minima are moderately accurate but not perfect. These points are marked in an image having intensity value zero throughout with one.

Detecting and localizing the hand extremities, that is, the fingertips and the valley between the fingers is the first step for hand geometry. Since both types of extremities are characterized by their high curvature, curve gram of the contour, that is, the plot of the curvature of the contour at various scales along the path length parameter was computed. The accuracy is high but with false points. Secondly points were extracted based on Euclidean distance. This have low accuracy but without any false points.

The proposed technique combines both approaches to get the best suited output. The algorithm is a decision based algorithm, Suppose X is the image obtained after Euclidian approach & Y is the image obtained after Corner approach. First locate the first white pixel in X. Scan the neighbourhood of the index of white pixel found in X in Y. If any white pixel is found then it is taken. Else the point originally found in X is taken.

2.4. Pose Correction

A landmark is a point of correspondence on each object that matches between and within populations. An example of a hand annotated with 28 landmarks is illustrated

Figure v: Landmark Points

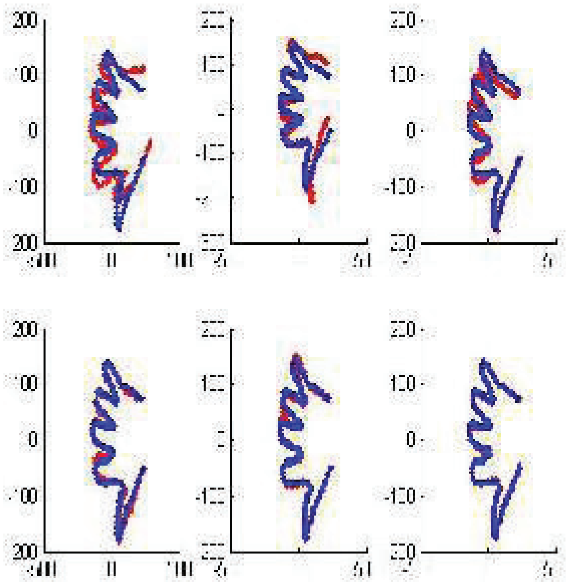

Figure vi: Pose corrected hand

For getting the landmark points-

1. Interpolate points threw the contour between the extremities
2. Draw line between these points
3. Find the distance of each line through the point
4. Using this distance information interpolate equally distance points
5. These points become the landmark points for comparison.

The database contains large number of hand. To obtain a true shape representation we have to filter out all the location, scale and rotational effects. Solution-establish a coordinate reference with respect to position, scale and rotation to which all the shapes will be aligned. This reference coordinate system is known as pose. From the database we select one image as target and remaining images has to be aligned to it.

First the average of the image is found and this is added to the coordinates to relocate the image to centre Average of the thickness is found and ratio of it with that of target is found using this we scale the image The angle formed between different points are found, The average of difference of angle from the target is used to correct the rotation.

2.5. Palm-Print Extraction

Palm-Print is the pixel intensities across the object in question (if necessary after a suitable normalization). For this mapping is done from one arbitrary point set $\{X_1 \cdots X_n\}$ into another $\{X'_1 \cdots X'_n\}$.

To partition the image for palm-print analysis, connect three landmark points in such a way that a triangle will be created. Then every point inside the triangle can be expressed by their relative distances from each of the vertices. This method is called Delaunay triangulation. Delaunay triangulation connects an irregular point set by a mesh of triangle's each satisfying the Delaunay property. The Delaunay property is as follows:

No triangle must have any point inside its circumcircle, which is the unique circle that connects all three points (vertices) of a triangle.

With implementation in mind, one should notice the common denominator. In pseudo code the piece-wise affine warp can be written as:

1. Determine the triangle, X belongs to
2. Find the relative position of X given by

$$X=\alpha \times a + \beta \times b + \gamma \times c \quad \text{(iii)}$$

$$\alpha = 1-(\beta + \gamma) \quad \text{(iv)}$$

$$\beta=\frac{yx_3-x_1y-x_3y_1-y_3x+x_1y_3+xy_1}{-x_2y_3+x_2y_1+x_1y_3+x_3y_2-x_3y_1-x_1y_2} \quad \text{(v)}$$

$$\gamma=\frac{yx_2-xy_1-x_1y_2-x_2y+x_2y_1+x_1y}{-x_2y_3+x_2y_1+x_1y_3+x_3y_2-x_3y_1-x_1y_2} \quad \text{(vi)}$$

Where $\alpha + \beta + \gamma = 1$

And X is inside a triangle if $0 \leq \alpha, \beta, \gamma \leq 1$

3. To obtain the mapped position

$$X'=f(X)=\alpha X'_1+\beta X'_2+\gamma X'_3 \quad \text{(vii)}$$

4. Set $I'(X)=I(f(x))$ (viii)
5. Interpolate the points between the obtained points

2.6. Databases

Here two databases are used instead of one for speeding up the operations. The two databases are maintained and classified based on left or right hand. To determine the hand to be detected is left or right

1. During the pose correction the target image to which the images has to be aligned should be held vertically.
2. Determine the coordinates of the extremities.
3. Select the two extremities in the rightmost and leftmost of the hand.
4. If the coordinates in the leftmost side is more than in the rightmost side then it is a right hand.
5. If the coordinates in the rightmost side is more than in the leftmost side then it is a left hand.
6. Based on the determined hand select the appropriate database.

3. Results

When using neighborhood scan based approach it can be seen that accuracy has been improved to a large extend. And can be verified by the following tables. X any are the coordinates.

Table i: Extremities determined using corner method

x	188	175	173	241	209
y	132	159	195	238	308

Table ii: Extremities determined using Euclidean method

x	111	188	71	175	55	173	74	241	197
y	108	132	144	159	188	195	217	238	259

Table iii: Extremities determined using Neighborhood method

x	111	189	71	177	55	168	74	239	197
y	108	134	144	166	188	206	217	250	259

It has been noted a 3% increase in the speed of seeking the data from the database if two database are used hence increasing the performance. The speed increase becomes much significance if we use a very large database. Overall accuracy of the system was found to be good with 76% accuracy in detecting the person.

4. Conclusions

By using neighborhood we obtained better detection of extremities. The multiple database technique had also provided significant speedup. Further improvement can be achieved by modifying the pose correction to work on three dimensional images and implementation more database based on more hand features for increased speed.

References

[1] R. Sanchez-Reillo, C. Sanchez-Avila, and A. Gonzalez-Macros, "Biometric identification through hand geometry measurements," *IEEETrans. Pattern Anal. Mach. Intell.*, vol. 22, no. 10, pp. 168–1171, Oct. 2000.

[2] A. K. Jain, A. Ross, and S. Pankanti, "A prototype hand geometrybased verification system," in *Proc. AVBPA*, Mar. 1999, pp. 166–171.

[3] S. Malassiotis, N. Aifanti, and M. G. Strintzis, "Personal authentication using 3-D finger geometry," *IEEE Trans. Inf. Forensics Security*, vol. 1, no. 1, pp. 12–21, Mar. 2006.

[4] N. Otsu, "A threshold selection method from gray-level histograms," *IEEE Trans. Syst., Man Cybernet.*, vol. 9, no. 1, pp. 62–66, Jan. 1979.

[5] W. Xiong, K. A. Toh, W. Y. Yau, and X. Jiang, "Model-guided deformable hand shape recognition without positioning aids," *PatternRecognit.*, vol. 38, no. 10, pp. 1651–1664, Oct. 2005.

[6] S. Ribaric and I. Fratric, "A biometric identification system based on eigenpalm and eigenfinger features," *IEEE Trans. Pattern Anal. Mach.Intell.*, vol. 27, no. 11, pp. 1698–1709, Nov. 2005.

[7] D. L. Woodard and P. J. Flynn, "Finger surface as a biometric identifier,"*Comput. Vis. Image Understand.*, vol. 100, no. 3, pp. 357–384, Dec. 2005.

[8] A. Kumar, "Incorporating cohort information for reliable palmprint authentication," in *Proc. ICVGIP*, Dec. 2008, pp. 583–590.

[9] C. Methani and A. M. Namboodiri, "Pose invariant palmprint recognition," in *Proc. ICB*, Jun. 2009, pp. 577–586.

[10] A. Morales, M. Ferrer, F. Díaz, J. Alonso, and C. Travieso, "Contactfree hand biometric system for real environments," in *Proc. 16th Eur.Signal Process. Conf.*, Laussane, Switzerland, Sep. 2008.

[11] V. Kanhangad, A. Kumar, and D. Zhang, "Combining 2-D and 3-D hand geometry features for biometric verification," in *Proc. IEEEWorkshop Biometrics*, Miami, FL, Jun. 2009, pp. 39–44. Xiao Chen He "Corner detector based on global and local curvature properties "*Optical Engineering"*